David's work

David Skrill

4/17/2021

```r
d <- readRDS("austin.rds")
d <- d %>% mutate(vehicle_year = ifelse(vehicle_year<1960,1960,vehicle_year)) %>%
  mutate(vehicle_year = ifelse(vehicle_year > 2017,NA, vehicle_year)) %>%
  mutate(vehicle_make = fct_lump_n(vehicle_make,n = 25)) %>%
  mutate(vehicle_model = fct_lump_n(vehicle_model,n = 250))
```

```r
d.numeric.all <- d %>% select(subject_age,subject_sex,frisk_performed,
                          search_conducted, search_person,
                          search_vehicle)
d.numeric.search <- d %>% filter(search_conducted==T) %>%
  select(contraband_found,contraband_drugs,contraband_weapons,frisk_performed)

d.cat <- d %>% select(subject_race,search_basis,reason_for_stop,
                      vehicle_make,vehicle_model,vehicle_registration_state,
                      vehicle_year)
```

```r
summary.stats1 <- d.numeric.all %>% mutate_all(as.numeric) %>%
  mutate(subject_sex = subject_sex - 1) %>%  #defaults to 1/2 coding
  as.data.frame() %>%
  fancy.summarize(.,nmis=T,uniq=T)
```

```
##                    nobs nmis uniq  mean    SD min 25% 50% 75% max
## subject_age      480091 3164   94 37.98 13.82  10  26  36  48 103
## subject_sex      482881  374    2  0.30  0.46   0   0   0   1   1
## frisk_performed  483255    0    2  0.02  0.15   0   0   0   0   1
## search_conducted 483255    0    2  0.04  0.20   0   0   0   0   1
## search_person    483255    0    2  0.03  0.18   0   0   0   0   1
## search_vehicle   483255    0    2  0.02  0.15   0   0   0   0   1
```

```r
summary.stats2 <- d.numeric.search %>%
  mutate_all(as.numeric) %>%
  as.data.frame() %>%
  fancy.summarize(uniq=T,nmis=T)
```

```
##                     nobs nmis uniq mean   SD min 25% 50% 75% max
## contraband_found   19256    0    2 0.25 0.43   0   0   0   0   1
## contraband_drugs   19256    0    2 0.01 0.12   0   0   0   0   1
## contraband_weapons 19256    0    2 0.05 0.21   0   0   0   0   1
## frisk_performed    19256    0    2 0.51 0.50   0   0   1   1   1
```

```r
summary.stats.tables <- apply(d.cat,2,tabyl)
summary.stats.tables
```

```
## $subject_race
```

```
##               newX[, i]      n      percent valid_percent
##  asian/pacific islander  13167 2.724648e-02   0.027246598
##                   black  72324 1.496601e-01   0.149660737
##                hispanic 123943 2.564754e-01   0.256476421
##                   other   2626 5.433984e-03   0.005434007
##                 unknown   3135 6.487258e-03   0.006487285
##                   white 268058 5.546927e-01   0.554694953
##                    <NA>      2 4.138602e-06            NA
##
## $search_basis
##       newX[, i]      n      percent valid_percent
##         consent   3195 0.0066114163   0.165922310
##           other    276 0.0005711270   0.014333195
##      plain view    152 0.0003145337   0.007893644
##  probable cause  15633 0.0323493808   0.811850852
##            <NA> 463999 0.9601535421            NA
##
## $reason_for_stop
##
##
##                                                                                           CALL I
##
##                                                                                         CALL FOI
##                                                                       CALL FOR SERVICE|PRE-EXISTING
##                                                                                      CALL FOR SER
##                                                                                                CA
##
##                                                                                              CONS
##                                                                                          CONSENSU
##
##                                                                                        CONSENSUAL
##                                                                                   CONSENSUAL CONT
##                                                                    CONSENSUAL CONTACT|SUSPICIOUS PI
##                                                              CONSENSUAL CONTACT|SUSPICIOUS PERSON /
##
##  NA|VIOLATION OF PENAL CODE|VIOLATION OF TRANSPORTATION/VEHICLE LAWS|PRE-EXISTING KNOWLEDGE|VIOLATIO
##    NA|VIOLATION OF TRANSPORTATION/VEHICLE LAWS|NA|CONSENSUAL CONTACT|VIOLATION OF TRANSPORTATION/VEHI
##
##
##
##
##                                                                                      PRE-EXISTING
##
##                                                                               PRE-EXISTING KNOWL
##
##                                                                                      SUSPICIOUS PI
##                                                            SUSPICIOUS PERSON / VEHICLE|CALL FOR SER
##                                                                                  SUSPICIOUS PERSO
##                                                                                                SU
##                                                                                SUSPICIOUS PERSON /
##                                                                          SUSPICIOUS PERSON / VEHIC
##                                                                           SUSPICIOUS PERSON / VEHI
##                                                                     SUSPICIOUS PERSON / VEHICLE|VI
##
```

```
##                                                                                    VIOLATION OF CIT
##                                                                                                   VI
##                                                                                  VIOLATION OF CITY (
##                                                                                        VIOLATION OF
##
##                                                                                            VIOLATIO
##                                                                                  VIOLATION OF PENAL
##                                                                                      VIOLATION OF P
##                                                                                 VIOLATION OF PENAL (
##                                                                                            VIOLATIO
##                                                                           VIOLATION OF TRANSPORTATI
##                                                         VIOLATION OF TRANSPORTATION/VEHICLE LAWS|CALL FO
##                                                             VIOLATION OF TRANSPORTATION/VEHICLE LAWS|CONS
##                                                                            VIOLATION OF TRANSPORTATION/V
##                                                                                       VIOLATION OF T
##                                                                          VIOLATION OF TRANSPORTATION/VEH
##                                                                     VIOLATION OF TRANSPORTATION/VEHICLE
##                                                    VIOLATION OF TRANSPORTATION/VEHICLE LAWS|SUSPICIOUS P
##                                                VIOLATION OF TRANSPORTATION/VEHICLE LAWS|SUSPICIOUS PERSO
##                                                                 VIOLATION OF TRANSPORTATION/VEHICLE LAWS|S
##                                              VIOLATION OF TRANSPORTATION/VEHICLE LAWS|SUSPICIOUS PERSON
##                                                                     VIOLATION OF TRANSPORTATION/VEHICLE
##                                                                         VIOLATION OF TRANSPORTATION/VEHI
##                                                                           VIOLATION OF TRANSPORTATI
##
##                                                                                                 WATER
##
##      n      percent valid_percent
##  17722 3.667215e-02  3.667253e-02
##      2 4.138602e-06  4.138645e-06
##      3 6.207903e-06  6.207967e-06
##      6 1.241581e-05  1.241593e-05
##      1 2.069301e-06  2.069322e-06
##      3 6.207903e-06  6.207967e-06
##      2 4.138602e-06  4.138645e-06
##    671 1.388501e-03  1.388515e-03
##      4 8.277204e-06  8.277289e-06
##      1 2.069301e-06  2.069322e-06
##      1 2.069301e-06  2.069322e-06
##      1 2.069301e-06  2.069322e-06
##     13 2.690091e-05  2.690119e-05
##      4 8.277204e-06  8.277289e-06
##      2 4.138602e-06  4.138645e-06
##    399 8.256511e-04  8.256596e-04
##      1 2.069301e-06  2.069322e-06
##      1 2.069301e-06  2.069322e-06
##  44717 9.253293e-02  9.253389e-02
##    104 2.152073e-04  2.152095e-04
##      1 2.069301e-06  2.069322e-06
##   1681 3.478495e-03  3.478531e-03
##      1 2.069301e-06  2.069322e-06
##      1 2.069301e-06  2.069322e-06
##      8 1.655441e-05  1.655458e-05
##  16528 3.420141e-02  3.420176e-02
```

```
##     110 2.276231e-04  2.276255e-04
##       2 4.138602e-06  4.138645e-06
##       1 2.069301e-06  2.069322e-06
##      14 2.897021e-05  2.897051e-05
##      18 3.724742e-05  3.724780e-05
##       1 2.069301e-06  2.069322e-06
##      21 4.345532e-05  4.345577e-05
##       1 2.069301e-06  2.069322e-06
##   43431 8.987181e-02  8.987274e-02
##       1 2.069301e-06  2.069322e-06
##       5 1.034650e-05  1.034661e-05
##       1 2.069301e-06  2.069322e-06
##       4 8.277204e-06  8.277289e-06
##    1169 2.419013e-03  2.419038e-03
##       1 2.069301e-06  2.069322e-06
##       1 2.069301e-06  2.069322e-06
##       1 2.069301e-06  2.069322e-06
##       1 2.069301e-06  2.069322e-06
##  353672 7.318538e-01  7.318614e-01
##      26 5.380182e-05  5.380238e-05
##       1 2.069301e-06  2.069322e-06
##       1 2.069301e-06  2.069322e-06
##      62 1.282967e-04  1.282980e-04
##      23 4.759392e-05  4.759441e-05
##      17 3.517812e-05  3.517848e-05
##      64 1.324353e-04  1.324366e-04
##       4 8.277204e-06  8.277289e-06
##       1 2.069301e-06  2.069322e-06
##       3 6.207903e-06  6.207967e-06
##       1 2.069301e-06  2.069322e-06
##      41 8.484134e-05  8.484221e-05
##     114 2.359003e-04  2.359027e-04
##       5 1.034650e-05  1.034661e-05
##    2551 5.278787e-03  5.278841e-03
##       3 6.207903e-06  6.207967e-06
##       5 1.034650e-05            NA
##
## $vehicle_make
##  newX[, i]     n    percent valid_percent
##       ACUR  5142 0.01064035    0.01092281
##        BMW  6071 0.01256273    0.01289622
##       BUIC  9118 0.01886789    0.01936876
##       CADI  6642 0.01374430    0.01410916
##       CHEV 53090 0.10985918    0.11277557
##       CHRY  7502 0.01552390    0.01593600
##       DODG 22238 0.04601711    0.04723871
##       FORD 55066 0.11394812    0.11697305
##       FRHT 18991 0.03929809    0.04034132
##        GMC  7352 0.01521350    0.01561737
##       HOND 35705 0.07388439    0.07584576
##       HYUN  8526 0.01764286    0.01811122
##       INTL 11170 0.02311409    0.02372769
##       JEEP  9417 0.01948661    0.02000391
##        KIA  7183 0.01486379    0.01525837
```

```
##       LEXS  6871 0.01421817    0.01459561
##       MACK  7043 0.01457409    0.01496098
##       MAZD 13650 0.02824596    0.02899579
##       MITS  6001 0.01241787    0.01274753
##       NISS 24609 0.05092343    0.05227527
##      Other 81300 0.16823416    0.17270020
##       PONT  6669 0.01380017    0.01416651
##       PTRB  8516 0.01762217    0.01808997
##       TOYT 39194 0.08110418    0.08325721
##       VOLK  7589 0.01570392    0.01612081
##       VOLV  6103 0.01262894    0.01296420
##       <NA> 12497 0.02586005            NA
##
## $vehicle_model
##  newX[, i]      n      percent valid_percent
##        200    131 0.0002710784  0.0008748614
##        240    150 0.0003103951  0.0010017497
##        28I     98 0.0002027915  0.0006544765
##       3 SE    130 0.0002690091  0.0008681831
##        300   1587 0.0032839805  0.0105985121
##        323    170 0.0003517812  0.0011353164
##        325    140 0.0002897021  0.0009349664
##        32I    102 0.0002110687  0.0006811898
##        330    175 0.0003621277  0.0011687080
##        350    102 0.0002110687  0.0006811898
##        4RN   1738 0.0035964449  0.0116069401
##        500    145 0.0003000486  0.0009683581
##        626    432 0.0008939380  0.0028850392
##         88     95 0.0001965836  0.0006344415
##         93     87 0.0001800292  0.0005810148
##        AA4    172 0.0003559198  0.0011486730
##        ACC   6147 0.0127199926  0.0410517036
##        AER     91 0.0001883064  0.0006077282
##        ALO    213 0.0004407611  0.0014224846
##        ALT   3253 0.0067314358  0.0217246123
##        AST    226 0.0004676620  0.0015093029
##        ATL    369 0.0007635720  0.0024643043
##        AVA    869 0.0017982225  0.0058034701
##        AVN    414 0.0008566906  0.0027648292
##        AVO    109 0.0002255538  0.0007279381
##        B23     79 0.0001634748  0.0005275882
##        B30     77 0.0001593362  0.0005142315
##        BLZ    604 0.0012498577  0.0040337122
##        BON    187 0.0003869593  0.0012488480
##        BRO    114 0.0002359003  0.0007613298
##        BUG    415 0.0008587599  0.0027715076
##        C/K     77 0.0001593362  0.0005142315
##        C15   1146 0.0023714188  0.0076533679
##        C23     81 0.0001676134  0.0005409449
##        C25    132 0.0002731477  0.0008815398
##        CAL    188 0.0003890286  0.0012555263
##        CAM   4609 0.0095374078  0.0307804298
##       CAMR     92 0.0001903757  0.0006144065
##        CAP    287 0.0005938894  0.0019166811
```

```
##         CAV    906 0.0018747866  0.0060505683
##         CBT    535 0.0011070760  0.0035729073
##         CCL    148 0.0003062565  0.0009883931
##         CEN    476 0.0009849872  0.0031788858
##         CHA   1098 0.0022720924  0.0073328080
##         CHK    938 0.0019410042  0.0062642749
##         CHL    254 0.0005256024  0.0016962962
##         CIV   5297 0.0109610868  0.0353751219
##          CL     82 0.0001696827  0.0005476232
##         CLD    154 0.0003186723  0.0010284630
##         CNT    117 0.0002421082  0.0007813648
##         COA   2932 0.0060671902  0.0195808679
##         CON    229 0.0004738699  0.0015293379
##         COR    124 0.0002565933  0.0008281131
##         COU    139 0.0002876328  0.0009282881
##         CPR    362 0.0007490869  0.0024175560
##         CRU    634 0.0013119368  0.0042340622
##        CRUZ    145 0.0003000486  0.0009683581
##         CRV   1454 0.0030087635  0.0097102940
##         CST    272 0.0005628498  0.0018165062
##         CTS    101 0.0002089994  0.0006745115
##         CUT    182 0.0003766128  0.0012154563
##         CVC    977 0.0020217070  0.0065247299
##         CVN   1005 0.0020796474  0.0067117231
##         CVT    165 0.0003414346  0.0011019247
##         CX7     86 0.0001779599  0.0005743365
##         DAK    445 0.0009208389  0.0029718575
##         DAR     89 0.0001841678  0.0005943715
##         DEV    895 0.0018520243  0.0059771067
##         DTS     97 0.0002007222  0.0006477982
##         DUR    697 0.0014423027  0.0046547970
##         E32     92 0.0001903757  0.0006144065
##         EC1    199 0.0004117909  0.0013289880
##         EC2     88 0.0001820985  0.0005876932
##         EC3     80 0.0001655441  0.0005342665
##         ECL    531 0.0010987988  0.0035461940
##         ECO    108 0.0002234845  0.0007212598
##         ECP    976 0.0020196377  0.0065180515
##         EDG    338 0.0006994237  0.0022572760
##         ELD     93 0.0001924450  0.0006210848
##         ELE    363 0.0007511562  0.0024242343
##         ELN    874 0.0018085690  0.0058368617
##         ENC     88 0.0001820985  0.0005876932
##         ENV    214 0.0004428304  0.0014291629
##         EPD   1450 0.0030004863  0.0096835807
##         EQX    436 0.0009022152  0.0029117525
##        ES 3    145 0.0003000486  0.0009683581
##         ESC    915 0.0018934103  0.0061106733
##         EXC    150 0.0003103951  0.0010017497
##         EXP    464 0.0009601556  0.0030987458
##         F15   5271 0.0109072850  0.0352014853
##        F150    250 0.0005173252  0.0016695829
##         F25    909 0.0018809945  0.0060706033
##         F35    341 0.0007056316  0.0022773110
```

```
##         FBD    135 0.0002793556  0.0009015748
##         FES    123 0.0002545240  0.0008214348
##         FIT    414 0.0008566906  0.0027648292
##         FJC     83 0.0001717520  0.0005543015
##         FLE     74 0.0001531283  0.0004941965
##         FOC   1699 0.0035157422  0.0113464852
##         FOR    404 0.0008359976  0.0026980459
##         FRT    597 0.0012353726  0.0039869639
##         FUS    760 0.0015726687  0.0050755319
##         G20     75 0.0001551976  0.0005008749
##         G35    426 0.0008815222  0.0028449692
##         G37     78 0.0001614055  0.0005209099
##          G6    363 0.0007511562  0.0024242343
##         GAL    693 0.0014340255  0.0046280837
##        GCAR     99 0.0002048608  0.0006611548
##         GCH   1104 0.0022845082  0.0073728780
##         GCK     85 0.0001758906  0.0005676582
##         GEN    100 0.0002069301  0.0006678331
##         GOL    199 0.0004117909  0.0013289880
##         GRA    385 0.0007966808  0.0025711576
##         GRM    719 0.0014878273  0.0048017203
##         GS3    137 0.0002834942  0.0009149314
##         GTI    125 0.0002586626  0.0008347914
##         GVT     75 0.0001551976  0.0005008749
##         HGH    412 0.0008525520  0.0027514726
##         HHR    285 0.0005897508  0.0019033245
##         HU2     77 0.0001593362  0.0005142315
##         I30    154 0.0003186723  0.0010284630
##         IMP   3311 0.0068514552  0.0221119555
##         INT    603 0.0012477884  0.0040270339
##         IS3    149 0.0003083258  0.0009950714
##         JET   1585 0.0032798419  0.0105851554
##         JMY    136 0.0002814249  0.0009082531
##         JNY    128 0.0002648705  0.0008548264
##         JUK     92 0.0001903757  0.0006144065
##         L40    121 0.0002503854  0.0008080781
##         LAN    317 0.0006559684  0.0021170311
##         LBY    601 0.0012436498  0.0040136772
##         LCR    275 0.0005690577  0.0018365412
##         LEG    231 0.0004780085  0.0015426946
##         LES   1184 0.0024500522  0.0079071445
##          LS    225 0.0004655927  0.0015026246
##         LUM    315 0.0006518298  0.0021036744
##          M3     78 0.0001614055  0.0005209099
##         MAG    202 0.0004179988  0.0013490230
##         MAL   1905 0.0039420182  0.0127222215
##         MAR    475 0.0009829179  0.0031722075
##         MAX   1425 0.0029487538  0.0095166224
##        MAZD    162 0.0003352267  0.0010818897
##         MDX    320 0.0006621763  0.0021370661
##         MIA    146 0.0003021179  0.0009750364
##         MIL     92 0.0001903757  0.0006144065
##         MIR    110 0.0002276231  0.0007346165
##         MOC    303 0.0006269982  0.0020235344
```

```
##         MON    306 0.0006332061  0.0020435694
##         MPV    131 0.0002710784  0.0008748614
##         MTN    173 0.0003579891  0.0011553513
##         MTX    128 0.0002648705  0.0008548264
##         MUR    296 0.0006125131  0.0019767861
##         MUS   2476 0.0051235890  0.0165355488
##         MX3    471 0.0009746407  0.0031454941
##         MX6    272 0.0005628498  0.0018165062
##         NAV    404 0.0008359976  0.0026980459
##         NEO    599 0.0012395112  0.0040003206
##         NIT    118 0.0002441775  0.0007880431
##         NOT    153 0.0003166030  0.0010217847
##         ODY    677 0.0014009167  0.0045212304
##         OPT    406 0.0008401362  0.0027114026
##       Other  14929 0.0308925929  0.0997008107
##         OUT     81 0.0001676134  0.0005409449
##         PAS    506 0.0010470662  0.0033792357
##         PAT     85 0.0001758906  0.0005676582
##         PCF     80 0.0001655441  0.0005342665
##         PLT    546 0.0011298383  0.0036463690
##         PRE    214 0.0004428304  0.0014291629
##         PRI   1412 0.0029218529  0.0094298041
##         PRK    484 0.0010015416  0.0032323124
##         PRO    680 0.0014071246  0.0045412654
##         PTH    748 0.0015478371  0.0049953920
##         Q45     91 0.0001883064  0.0006077282
##         QST    161 0.0003331574  0.0010752114
##         R15   2048 0.0042379282  0.0136772229
##         R25    177 0.0003662663  0.0011820647
##         R35     99 0.0002048608  0.0006611548
##         RAM    251 0.0005193945  0.0016762612
##         RAV    668 0.0013822930  0.0044611254
##         RDV     72 0.0001489897  0.0004808399
##         REG    388 0.0008028887  0.0025911926
##         RGE    323 0.0006683842  0.0021571011
##         RGL     82 0.0001696827  0.0005476232
##         RIO    425 0.0008794529  0.0028382909
##         RNG   1168 0.0024169434  0.0078002912
##         ROA    108 0.0002234845  0.0007212598
##         ROD    409 0.0008463441  0.0027314376
##         RRV    210 0.0004345532  0.0014024496
##         RSX    146 0.0003021179  0.0009750364
##        RX 3    125 0.0002586626  0.0008347914
##         RX3    295 0.0006104438  0.0019701078
##         S10    412 0.0008525520  0.0027514726
##         S40    151 0.0003124644  0.0010084281
##         S60    174 0.0003600584  0.0011620297
##         S80     72 0.0001489897  0.0004808399
##         SAB    206 0.0004262760  0.0013757363
##         SEB    594 0.0012291647  0.0039669289
##         SED    146 0.0003021179  0.0009750364
##         SEN   1349 0.0027914869  0.0090090692
##         SEP     80 0.0001655441  0.0005342665
##         SEV    138 0.0002855635  0.0009216097
```

```
##        SFE    364 0.0007532255  0.0024309127
##       SILV    239 0.0004945629  0.0015961212
##         SL    190 0.0003931672  0.0012688830
##        SLV   3060 0.0063320607  0.0204356943
##        SNA    449 0.0009291161  0.0029985708
##        SNF    183 0.0003786821  0.0012221347
##        SOL    466 0.0009642942  0.0031121025
##        SON    999 0.0020672316  0.0066716532
##        SOR    315 0.0006518298  0.0021036744
##        SPC    425 0.0008794529  0.0028382909
##        SPE     85 0.0001758906  0.0005676582
##        SPT    474 0.0009808486  0.0031655291
##        SQA    237 0.0004904243  0.0015827646
##        SRA    977 0.0020217070  0.0065247299
##        SRX    135 0.0002793556  0.0009015748
##        STA    584 0.0012084717  0.0039001456
##        STS     97 0.0002007222  0.0006477982
##        SUB   2090 0.0043248389  0.0139577128
##        TAC   1025 0.0021210334  0.0068452898
##        TAH   2295 0.0047490455  0.0153267708
##        TAU   1697 0.0035116036  0.0113331285
##        TBZ    823 0.0017030346  0.0054962668
##         TC    225 0.0004655927  0.0015026246
##        TCN    120 0.0002483161  0.0008013998
##        TER     98 0.0002027915  0.0006544765
##        THU    120 0.0002483161  0.0008013998
##        TIB    125 0.0002586626  0.0008347914
##        TOW   1024 0.0021189641  0.0068386114
##        TRB    375 0.0007759878  0.0025043743
##        TRP    124 0.0002565933  0.0008281131
##        TRV     71 0.0001469204  0.0004741615
##        TSX     95 0.0001965836  0.0006344415
##        TTN    200 0.0004138602  0.0013356663
##        TUN    984 0.0020361921  0.0065714782
##        V15     78 0.0001614055  0.0005209099
##        V70     82 0.0001696827  0.0005476232
##        VEN     85 0.0001758906  0.0005676582
##        VIB    203 0.0004200681  0.0013557013
##        VOY    153 0.0003166030  0.0010217847
##        VRS    443 0.0009167003  0.0029585008
##        VUE    195 0.0004035137  0.0013022746
##        WIN    193 0.0003993751  0.0012889180
##        WRG    662 0.0013698772  0.0044210554
##         X3     79 0.0001634748  0.0005275882
##         X5    156 0.0003228109  0.0010418197
##        XPL   2721 0.0056305677  0.0181717400
##        XST    109 0.0002255538  0.0007279381
##        XTR    770 0.0015933617  0.0051423152
##        YAR    332 0.0006870079  0.0022172061
##        YUK    750 0.0015519757  0.0050087486
##       <NA> 333517 0.6901470238            NA
##
## $vehicle_registration_state
##  newX[, i]      n      percent valid_percent
```

```
##          AK    102 2.110687e-04  2.143042e-04
##          AL    602 1.245719e-03  1.264815e-03
##          AR    760 1.572669e-03  1.596776e-03
##          AZ    756 1.564391e-03  1.588372e-03
##          CA   2501 5.175322e-03  5.254654e-03
##          CO    638 1.320214e-03  1.340452e-03
##          CT    112 2.317617e-04  2.353144e-04
##          DC     28 5.794042e-05  5.882860e-05
##          DE     31 6.414833e-05  6.513166e-05
##          FL   1837 3.801306e-03  3.859576e-03
##          GA    831 1.719589e-03  1.745949e-03
##          HI     40 8.277204e-05  8.404085e-05
##          IA    699 1.446441e-03  1.468614e-03
##          ID    107 2.214152e-04  2.248093e-04
##          IL   2595 5.369836e-03  5.452150e-03
##          IN   2403 4.972530e-03  5.048754e-03
##          KS    535 1.107076e-03  1.124046e-03
##          KY    286 5.918201e-04  6.008921e-04
##          LA   1668 3.451594e-03  3.504504e-03
##          MA    226 4.676620e-04  4.748308e-04
##          MD    215 4.448997e-04  4.517196e-04
##          ME   1219 2.522478e-03  2.561145e-03
##          MI    794 1.643025e-03  1.668211e-03
##          MN    525 1.086383e-03  1.103036e-03
##          MO   1137 2.352795e-03  2.388861e-03
##          MS    506 1.047066e-03  1.063117e-03
##          MT    128 2.648705e-04  2.689307e-04
##          NC    692 1.431956e-03  1.453907e-03
##          ND     86 1.779599e-04  1.806878e-04
##          NE    496 1.026373e-03  1.042107e-03
##          NH     72 1.489897e-04  1.512735e-04
##          NJ    270 5.587112e-04  5.672758e-04
##          NM    641 1.326422e-03  1.346755e-03
##          NV    201 4.159295e-04  4.223053e-04
##          NY    382 7.904729e-04  8.025901e-04
##          OH    831 1.719589e-03  1.745949e-03
##          OK   4910 1.016027e-02  1.031601e-02
##          OR    464 9.601556e-04  9.748739e-04
##          PA    417 8.628985e-04  8.761259e-04
##          RI     40 8.277204e-05  8.404085e-05
##          SC    286 5.918201e-04  6.008921e-04
##          SD    113 2.338310e-04  2.374154e-04
##          TN   2551 5.278787e-03  5.359705e-03
##          TX 440405 9.113305e-01  9.253003e-01
##          UT    215 4.448997e-04  4.517196e-04
##          VA    437 9.042845e-04  9.181463e-04
##          VT     26 5.380182e-05  5.462655e-05
##          WA    370 7.656413e-04  7.773779e-04
##          WI    685 1.417471e-03  1.439200e-03
##          WV     42 8.691064e-05  8.824289e-05
##          WY     46 9.518784e-05  9.664698e-05
##        <NA>   7296 1.509762e-02            NA
##
## $vehicle_year
```

```
##   newX[, i]     n      percent valid_percent
##        1960    68 1.407125e-04  1.825043e-04
##        1961     8 1.655441e-05  2.147109e-05
##        1962    11 2.276231e-05  2.952275e-05
##        1963    25 5.173252e-05  6.709716e-05
##        1964    20 4.138602e-05  5.367773e-05
##        1965    34 7.035623e-05  9.125214e-05
##        1966    56 1.158808e-04  1.502976e-04
##        1967    43 8.897994e-05  1.154071e-04
##        1968    54 1.117422e-04  1.449299e-04
##        1969    42 8.691064e-05  1.127232e-04
##        1970    67 1.386432e-04  1.798204e-04
##        1971    58 1.200195e-04  1.556654e-04
##        1972    87 1.800292e-04  2.334981e-04
##        1973    93 1.924450e-04  2.496014e-04
##        1974   142 2.938407e-04  3.811119e-04
##        1975    66 1.365739e-04  1.771365e-04
##        1976   106 2.193459e-04  2.844920e-04
##        1977   185 3.828207e-04  4.965190e-04
##        1978   231 4.780085e-04  6.199778e-04
##        1979   288 5.959587e-04  7.729593e-04
##        1980   268 5.545726e-04  7.192816e-04
##        1981   303 6.269982e-04  8.132176e-04
##        1982   390 8.070273e-04  1.046716e-03
##        1983   391 8.090966e-04  1.049400e-03
##        1984   752 1.556114e-03  2.018283e-03
##        1985   869 1.798222e-03  2.332297e-03
##        1986   980 2.027915e-03  2.630209e-03
##        1987  1123 2.323825e-03  3.014005e-03
##        1988  1419 2.936338e-03  3.808435e-03
##        1989  2103 4.351740e-03  5.644213e-03
##        1990  2673 5.531241e-03  7.174029e-03
##        1991  3334 6.899049e-03  8.948078e-03
##        1992  4103 8.490342e-03  1.101199e-02
##        1993  5703 1.180122e-02  1.530620e-02
##        1994  7346 1.520108e-02  1.971583e-02
##        1995  9282 1.920725e-02  2.491183e-02
##        1996  9645 1.995841e-02  2.588609e-02
##        1997 11974 2.477781e-02  3.213686e-02
##        1998 14347 2.968826e-02  3.850572e-02
##        1999 17453 3.611551e-02  4.684187e-02
##        2000 21106 4.367466e-02  5.664611e-02
##        2001 20447 4.231100e-02  5.487743e-02
##        2002 21056 4.357120e-02  5.651191e-02
##        2003 21813 4.513766e-02  5.854362e-02
##        2004 21913 4.534459e-02  5.881200e-02
##        2005 24646 5.099999e-02  6.614707e-02
##        2006 26423 5.467714e-02  7.091633e-02
##        2007 28365 5.869572e-02  7.612844e-02
##        2008 20930 4.331047e-02  5.617374e-02
##        2009 13164 2.724028e-02  3.533068e-02
##        2010 12603 2.607940e-02  3.382502e-02
##        2011 10615 2.196563e-02  2.848946e-02
##        2012 11052 2.286991e-02  2.966231e-02
```

```
##       2013   9609 1.988391e-02  2.578947e-02
##       2014   7169 1.483482e-02  1.924078e-02
##       2015   4437 9.181488e-03  1.190840e-02
##       2016   1078 2.230706e-03  2.893230e-03
##       2017     26 5.380182e-05  6.978105e-05
##       <NA> 110661 2.289909e-01            NA
```

Investigating the Hit Rate

```r
d %>% filter(search_conducted==T) %>% tabyl(subject_race)
```

```
##            subject_race    n     percent
##  asian/pacific islander  191 0.009918986
##                   black 5071 0.263346489
##                hispanic 7057 0.366483174
##                   white 6774 0.351786456
##                   other   73 0.003791026
##                 unknown   90 0.004673868
```

Because there are relatively few stops of Asian/Pacific Islanders or people of other races, I will combine these. Since we are primarilly interested in the differences in hit rates by race, I will remove the relatively few instances in which race is unknown.

```r
d2 <- d %>% filter(subject_race != "unknown") %>%
  mutate(subject_race = fct_lump_min(subject_race,20000))
```

```r
d2 %>%
  filter(search_conducted) %>%
  group_by(subject_race) %>%
  summarise(
    hit_rate = mean(contraband_found, na.rm = T)
  )
```

```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 4 x 2
##   subject_race hit_rate
##   <fct>           <dbl>
## 1 black           0.252
## 2 hispanic        0.260
## 3 white           0.238
## 4 Other           0.152
```

```r
d2 %>%
  filter(search_conducted) %>%
  group_by(subject_race) %>%
  summarise(
    hit_rate = mean(frisk_performed, na.rm = T)
  )
```

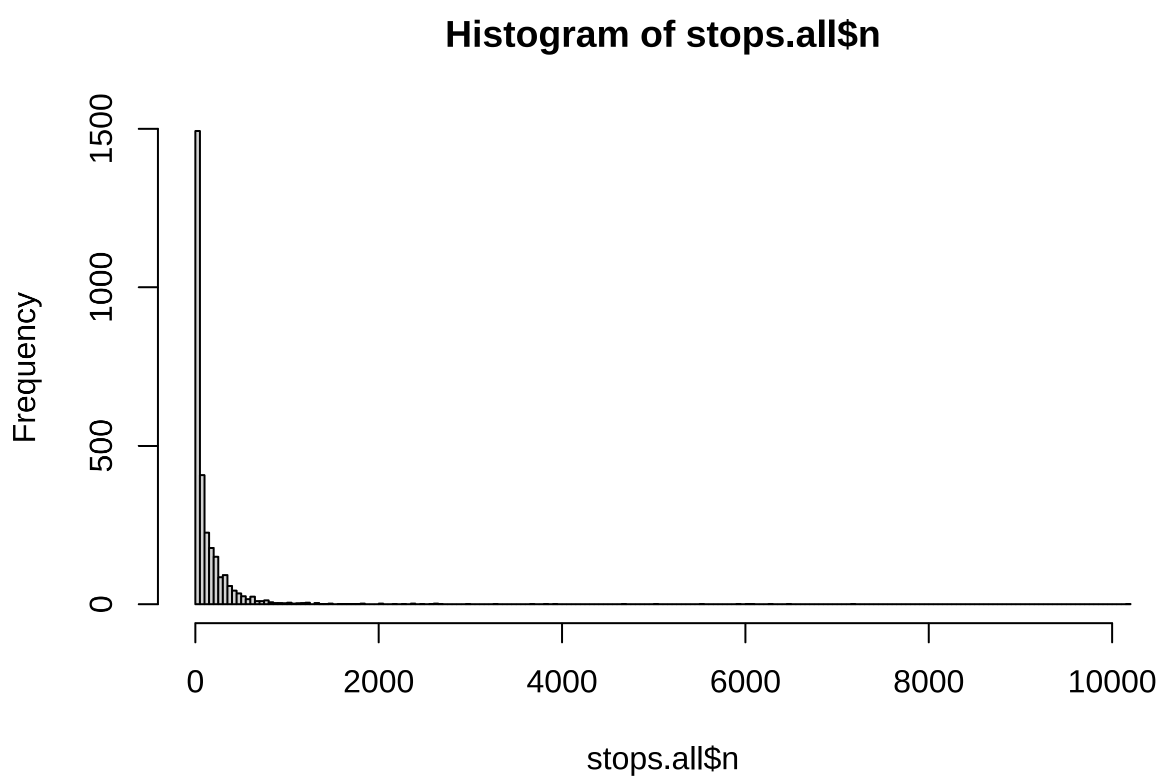
```
## `summarise()` ungrouping output (override with `.groups` argument)

## # A tibble: 4 x 2
##   subject_race hit_rate
##   <fct>           <dbl>
## 1 black           0.460
## 2 hispanic        0.525
```

```
## 3 white          0.542
## 4 Other          0.549
```

I will investigate the hit rate at the individual officer level. There are many officers with very few (or only one) stop. There is a very long tail, with one officer making over 10000 stops (averaging over 3 per day over the study period)! There is a similar pattern in the number of searches conducted. I will restrict the analysis to officers with 12 or more searches (corresponding to officers in or above the 75th percentile).

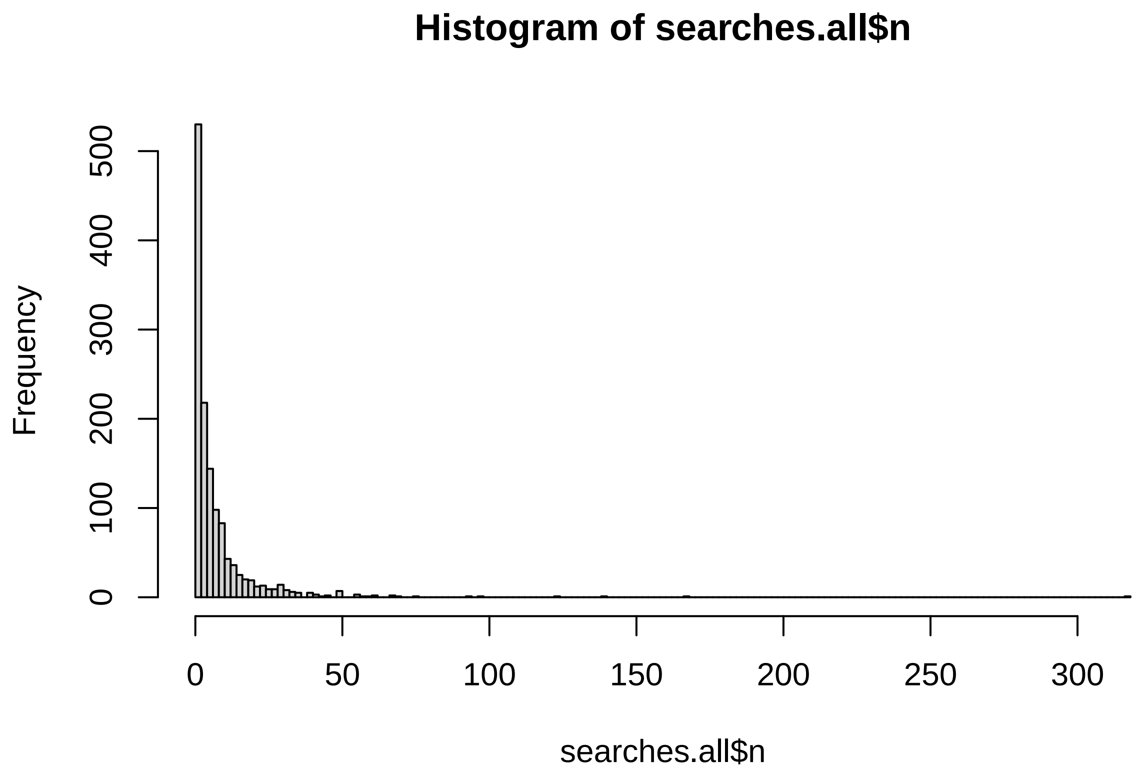
```
stops.all <- d2 %>% tabyl(officer_id_hash)
hist(stops.all$n,breaks=200)
```




```
quantile(stops.all$n)
```

```
##    0%   25%   50%   75%  100%
##     1     6    48   170 10197
```

```
searches.all <- d2 %>% filter(frisk_performed==T) %>%
  tabyl(officer_id_hash)
hist(searches.all$n,breaks=200)
```




```
quantile(searches.all$n)
```

```
##   0%  25%  50%  75% 100%
##    1    1    4    9  318
```

```
quantile(searches.all$n,seq(0,1,.1))
```

```
##   0%  10%  20%  30%  40%  50%  60%  70%  80%  90% 100%
##    1    1    1    2    3    4    5    7   10   18  318
```

```
ids.to.keep <- searches.all %>% filter(n>18) %>% pull(officer_id_hash)
searches <- d2 %>% filter(officer_id_hash %in% ids.to.keep) %>%
  filter(search_conducted==T) %>%
  mutate(hit = (contraband_found | contraband_drugs | contraband_weapons)) %>%
  select(officer_id_hash, contains("subject_"),
         reason_for_stop,frisk_performed,search_conducted,hit,contains("contraband"))
head(searches)
```

```
## # A tibble: 6 x 11
##   officer_id_hash subject_age subject_race subject_sex reason_for_stop
##   <chr>                 <int> <fct>        <fct>       <chr>
## 1 7141a6e62f               23 hispanic     female      VIOLATION OF T~
## 2 2eafe65a93               21 white        female      VIOLATION OF T~
## 3 4e247875fc               30 hispanic     male        VIOLATION OF T~
## 4 1ca4754b63               21 white        male        VIOLATION OF T~
## 5 19edc7fef5               55 white        male        VIOLATION OF T~
## 6 19edc7fef5               25 hispanic     male        VIOLATION OF T~
## # ... with 6 more variables: frisk_performed <lgl>, search_conducted <lgl>,
```

```
## #   hit <lgl>, contraband_found <lgl>, contraband_drugs <lgl>,
## #   contraband_weapons <lgl>
```

```r
# hit_rates <- searches %>%
#   group_by(officer_id_hash,subject_race) %>%
#   summarise(
#     hit_rate = mean(contraband_found, na.rm = T)
#   )
hit_rates <- searches %>%
  group_by(officer_id_hash,subject_race) %>%
  summarise(
    hit_rate = mean(frisk_performed, na.rm = T)
  )
```

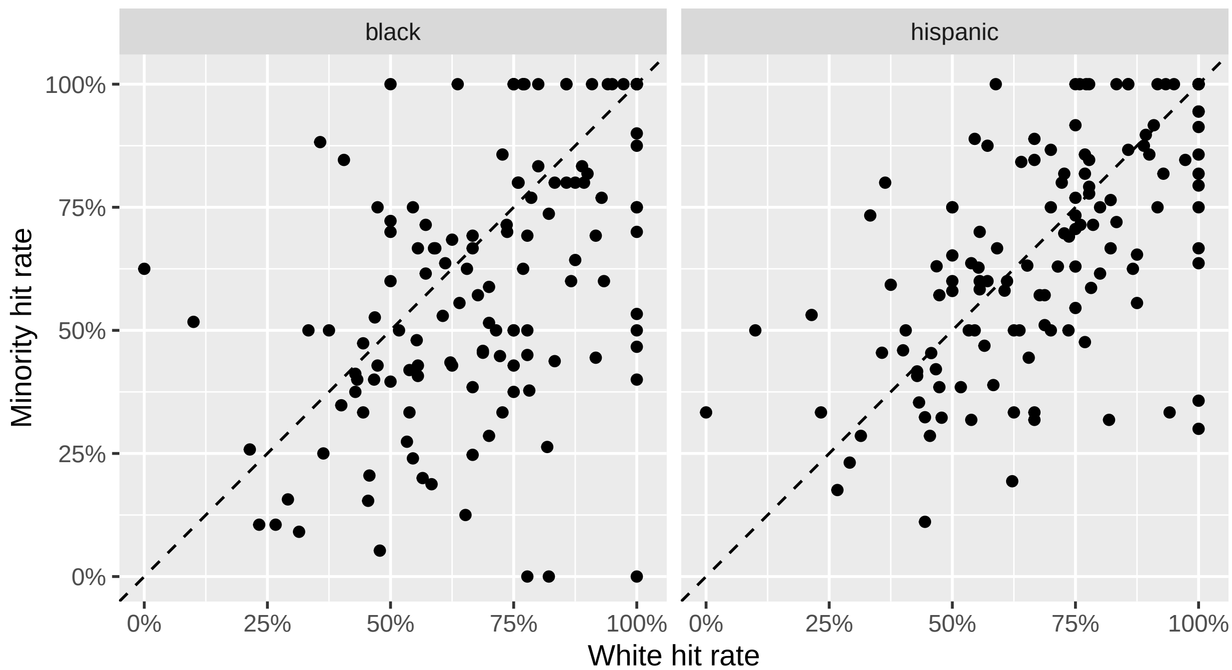
```
## `summarise()` regrouping output by 'officer_id_hash' (override with `.groups` argument)
```

```r
hit_rates <- hit_rates %>%
  filter(subject_race %in% c("black", "white", "hispanic")) %>%
  spread(subject_race, hit_rate, fill = 0) %>%
  rename(white_hit_rate = white) %>%
  gather(minority_race, minority_hit_rate, c(black, hispanic)) %>%
  arrange(officer_id_hash)
hit_rates
```

```
## # A tibble: 260 x 4
## # Groups:   officer_id_hash [130]
##    officer_id_hash white_hit_rate minority_race minority_hit_rate
##    <chr>                    <dbl> <chr>                     <dbl>
##  1 01db7098a7               0.857 black                     0.8
##  2 01db7098a7               0.857 hispanic                  0.867
##  3 020579eaad               0.606 black                     0.529
##  4 020579eaad               0.606 hispanic                  0.581
##  5 02b0803fe3               0.429 black                     0.412
##  6 02b0803fe3               0.429 hispanic                  0.407
##  7 0329f48f95               0.375 black                     0.5
##  8 0329f48f95               0.375 hispanic                  0.593
##  9 068ff01d47               0.517 black                     0.5
## 10 068ff01d47               0.517 hispanic                  0.385
## # ... with 250 more rows
```

```r
# We'll use this just to make our axes' limits nice and even
max_hit_rate <- hit_rates %>% ungroup %>%
  select(ends_with("hit_rate")) %>%
  max()
hit_rates %>%
  ggplot(aes(
    x = white_hit_rate,
    y = minority_hit_rate
  )) +
  geom_point() +
  # This sets a diagonal reference line (line of equal hit rates)
  geom_abline(slope = 1, intercept = 0, linetype = "dashed") +
  # These next few lines just make the axes pretty and even
  scale_x_continuous("White hit rate",
    limits = c(0, max_hit_rate + 0.01),
    labels = scales::percent
```

```r
  ) +
  scale_y_continuous("Minority hit rate",
    limits = c(0, max_hit_rate + 0.01),
    labels = scales::percent
  ) +
  # This makes sure that 1% on the x-axis is the same as 1% on the y-axis
  coord_fixed() +
  # This allows us to compare black v. white and Hispanic v. white side by
  # side, in panels
  facet_grid(. ~ minority_race)
```




```r
  # Depending on your version of ggplot2, you may be able to use the syntax
  # below (the newer ggplot2 syntax)---which is clearer, in my opinion.
  # But older versions of ggplot2 will only accept the above syntax
  # facet_grid(cols = vars(minority_race))
```

```r
searches <- d2 %>% filter(officer_id_hash %in% ids.to.keep) %>%
  filter(subject_race %in% c("black","hispanic","white")) %>%
  mutate(subject_race = relevel(subject_race,ref="white")) %>%
  mutate(subject_race = fct_drop(subject_race)) %>%
  filter(frisk_performed==T) %>%
  mutate(hit = (contraband_found | contraband_drugs | contraband_weapons)) %>%
  select(officer_id_hash, contains("subject_"),
         reason_for_stop,frisk_performed,search_conducted,hit,contains("contraband"))

hit_rates_binary <- searches %>%
```

```r
  filter(subject_race %in% c("black","hispanic","white")) %>%
  mutate(subject_race = fct_drop(subject_race)) %>%
  mutate(subject_race = relevel(subject_race,ref="white")) %>%
  group_by(officer_id_hash,subject_race) %>%
  summarise(
    hit_rate = mean(contraband_found, na.rm = T),
    nsearches = n(),
    nhits = sum(contraband_found,na.rm=T)
  )
```

```
## `summarise()` regrouping output by 'officer_id_hash' (override with `.groups` argument)
```

```r
table(hit_rates_binary$subject_race)
```

```
##
##    white    black hispanic
##      129      127      130
```

```r
library(rstanarm)
```

```
## Loading required package: Rcpp
## This is rstanarm version 2.21.1
## - See https://mc-stan.org/rstanarm/articles/priors for changes to default priors!
## - Default priors may change, so it's safest to specify priors, even if equivalent to the defaults.
## - For execution on a local, multicore CPU with excess RAM we recommend calling
##   options(mc.cores = parallel::detectCores())
```

```r
summary_stats <- function(posterior) {
  x <- invlogit(posterior)  # log-odds -> probabilities
  t(apply(x, 2, quantile, probs = c(0.025, 0.5, 0.975)))
}
shift_draws <- function(draws) {
  sweep(draws[, -1], MARGIN = 1, STATS = draws[, 1], FUN = "+")
}
```

```r
SEED <- 101
wi_prior <- normal(-1.3, 1) #the overall hit rate is .298; log(.298) ~ -1.21
stanfit1 <- stan_glmer(cbind(nhits, nsearches-nhits) ~ 1 + (1 +subject_race | officer_id_hash),
                              data = hit_rates_binary, family = binomial("logit"),
             prior_intercept = wi_prior, seed = SEED)
```

```r
stanfit1 <- readRDS("stanfit1.rds")
alphas <- shift_draws(as.matrix(stanfit1))
post1 <- summary_stats(alphas)
post1 <- post1[-nrow(post1),]
```

```r
officer.medians <- post1[1:(nrow(post1)-5), 2]
median.mat <- matrix(officer.medians,nrow=length(unique(ids.to.keep)),
                          ncol=3,byrow = T)
mu <- apply(median.mat,1,mean)
output = rep(0,130)
for (ii in 1:130){
  temp = mean(median.mat[ii,])
  ssr = sum((median.mat[ii,] - mu[ii])^2)
```

```
  output[ii] = ssr
}
```

```
# fit2 <- stan_glmer(cbind(nhits, nsearches-nhits) ~ (1 | officer_id_hash),
#                               data = hit_rates_binary, family = binomial("logit"),
#              prior_intercept = wi_prior, seed = SEED)
# alphas <- shift_draws(as.matrix(fit2))
# partialpool <- summary_stats(alphas)
# partialpool <- partialpool[-nrow(partialpool),]
#
# fit3 <- stan_glmer(cbind(nhits, nsearches-nhits) ~ subject_race + (1 | officer_id_hash),
#                               data = hit_rates_binary, family = binomial("logit"),
#              prior_intercept = wi_prior, seed = SEED)
# alphas <- shift_draws(as.matrix(fit3))
# partialpool <- summary_stats(alphas)
# partialpool <- partialpool[-nrow(partialpool),]}
# SEED <- 101
# wi_prior <- normal(-1.2, 1) #the overall hit rate is .298; log(.298) ~ -1.21
# fit_partialpool <-stan_glmer(cbind(nhits, nsearches-nhits) ~ (subject_race | officer_id_hash),
#                               data = hit_rates_binary, family = binomial("logit"),
#              prior_intercept = wi_prior, seed = SEED)
```

```
# logit1 <- glm(hit ~ subject_race, data = searches, family="binomial")
# summary(logit1)
# ```
#
# It is likely the case that the hit rate will vary substantially by officer.
# ```{r}
# lmer1 <- lme4::glmer(hit ~ 1 + (1|subject_race),
#                       data = searches, family="binomial")
# summary(lmer1)
# ```
#
# ```{r}
# lmer2 <- lme4::glmer(hit ~1 +  subject_race + (1|officer_id_hash),
#                       data = searches, family="binomial")
# summary(lmer2)
# ```
#
# ```{r}
# lmer3 <- lme4::glmer(hit ~1 + (1 +subject_race | officer_id_hash),
#                       data = searches, family="binomial")
# summary(lmer3)
```